IEEE CSS

Received 26 August 2022; revised 26 November 2022; accepted 19 December 2022. Date of publication 9 January 2023; date of current version 31 January 2023. Recommended by Senior Editor Prof. Wei Zhang.

Digital Object Identifier 10.1109/OJCSYS.2023.3235248

Cross Apprenticeship Learning Framework: Properties and Solution Approaches

ASHWIN ARAVIND [1] (Graduate Student Member, IEEE), DEBASISH CHATTERJEE [1] (Senior Member, IEEE), AND ASHISH CHERUKURI [2] (Member, IEEE)

[1]Department of Systems and Control Engineering, Indian Institute of Technology Bombay, Mumbai 400076, India
[2]Engineering and Technology Institute Groningen, University of Groningen, Nijenborgh 4, 9747 AG Groningen, The Netherlands

CORRESPONDING AUTHOR: ASHWIN ARAVIND (e-mail: ashwinaravind1@gmail.com)

This work was supported by the Ministry of Human Resource Development, Govt. of India.

ABSTRACT Apprenticeship learning is a framework in which an agent learns a policy to perform a given task in an environment using example trajectories provided by an expert. In the real world, one might have access to expert trajectories in different environments where system dynamics is different while the learning task is the same. For such scenarios, two types of learning objectives can be defined. One where the learned policy performs very well in one specific environment and another when it performs well across all environments. To balance these two objectives in a principled way, our work presents the cross apprenticeship learning (CAL) framework. This consists of an optimization problem where an optimal policy for each environment is sought while ensuring that all policies remain close to each other. This nearness is facilitated by one tuning parameter in the optimization problem. We derive properties of the optimizers of the problem as the tuning parameter varies. We identify conditions under which an agent prefers using the policy obtained from CAL over the traditional apprenticeship learning. Since the CAL problem is nonconvex, we provide a convex outer approximation. Finally, we demonstrate the attributes of our framework in the context of a navigation task in a windy gridworld environment.

INDEX TERMS Apprenticeship learning, multiagent systems, reinforcement learning, stochastic control.

I. INTRODUCTION

Reinforcement learning involves learning via interaction with the environment to perform a task optimally in a sequential decision-making process [1]. Commonly, the agent takes an action at a state, transitions to another state, obtains a reward from the environment and repeats the whole process again. Learning occurs when the agent looks to maximize the long-term reward, and so the efficacy of learning relies heavily on the reward structure. Poorly defined rewards lead to unwanted behaviour. In several control applications, defining appropriate rewards is difficult, and most likely, the desired behaviour can be demonstrated by an expert. For such cases, several methods under the broad umbrella of *learning from demonstrations* are studied in the past, where the behaviour of an expert is available in terms of state-action trajectories [2]. This information can be used in different ways, out of which, a common one is the framework of *apprenticeship learning* [3]. Here, the goal is to recover optimal policies for a given Markov decision process (MDP) using demonstrations from the expert and the fact that the set where the reward function belongs to is known.

In real life, we envision scenarios where trajectories from multiple experts in different environments are available, but the underlying task is common across environments. In such settings, an agent in an environment can learn a policy that seeks a trade-off between its performance in its own environment and across multiple environments. The former is attractive when the agent is supposed to only operate in its own environment and the available trajectories from its own expert are sufficient. The latter is advantageous in cases where the learned policy is supposed to work well across a range of environments and also possibly work as a warm start for specialized learning in any particular environment. Taking these considerations as the motivation, we present the *cross apprenticeship learning* (CAL) framework in this work. We analyze the properties of the policies obtained from this framework

and then address computational issues. We also show that our framework is beneficial, for example, when one agent has access to imperfect expert behavior or when agents need to perform the task in unseen environments.

A. LITERATURE REVIEW

Apprenticeship learning, as introduced in [4] consisted of two steps, first was to infer the reward function governing the expert's actions using inverse reinforcement learning, and the second was learning a suitable optimal policy for this reward function using reinforcement learning. Here, although the reward function was unknown to the agent, it was known to belong to the set of linear combinations of certain basis vectors. The applications of this framework are plenty, for example, to learn aerobatic manoeuvres on a helicopter [5], [6], quadrupled locomotion [7], navigation in a parking lot [8], and automated parking [9]. In [10] a game-theoretic approach for apprenticeship learning was proposed: the problem was cast as a two-player zero-sum game where the learning agent chooses the policy and the environment selects the reward function. This framework resulted in computationally inexpensive method and it found policies that are guaranteed to be at least as good as the expert policy for any given reward function. Building on [10] and the linear programming (LP) approach for finding optimal policies given in [11], the work [3] proposes an LP formulation for apprenticeship learning. The work [12], motivated by [13], extended this LP framework to large-scale problems by solving an approximate problem where the decision variable is assumed to lie in a subspace generated by feature vectors.

In our work, we use the LP framework for apprenticeship learning as a starting point. Our objective in this article differs from the above mentioned methods because we wish to learn a policy that is able to perform a task well in multiple environments by exploiting the availability of expert demonstrations in these environments. Such policies have a definite edge in terms of robustness as compared to policies that are learned in only one environment.

Closely related to our work are [14], [15] and [16]. The work [14] aims to find a policy that performs well in different scenarios of an MDP where the scenarios are supposed to be representative of the change in the agent's environment. We note that the setting is not of learning from an expert. Instead, they assume that the reward function is given. In [15], expert demonstrations from different environments, parameterized by a context variable, are used to infer a parameterized reward function. The aim is to use the inferred function to perform learning in unseen environments. The work [16] explores a similar setup for imitation learning as ours. Here, minimization of the Jensen-Shannon divergence between the agent's policy and the experts' policies in different environments improved robustness to variations in environment dynamics compared to baseline imitation learning techniques. Unlike these methods, we use the LP-based approach to define cross-learning, where we borrow the key ideas of centrality of policies from [17] to find a middle ground between performance in one single environment and performance in all environments.

Apart from these works, there is a growing interest in inverse reinforcement or imitation learning for linear systems [18], nonlinear systems [19], [20], [21] and MDPs [22], [23], [24].

B. SETUP AND CONTRIBUTIONS

For a single-agent single-environment case, the apprenticeship learning framework involves finding a policy that minimizes the worst-case discrepancy between the cost incurred by the said policy and an expert policy. In here, the worst-case discrepancy is computed by considering all cost functions that belong to a linear subspace spanned by a certain number of basis vectors. This is motivated by the setting where the learning agent does not have access to the actual cost function driving the expert behaviour but knows the set where it belongs to. This worst-case minimization problem can be cast as an LP in terms of the occupation measure. It is assumed that the learning agent does not have access to the policy of the expert. Instead, the occupation measure corresponding to the expert policy is available as it can be easily approximated using available expert trajectories.

Our *first contribution* proposes the CAL framework that extends the above defined single agent apprenticeship learning to multiple agents. At the core of this framework is the optimization problem where we seek a policy for each environment that balances two objectives. First, it minimizes the worst-case discrepancy measure, as explained above for a single agent case, for its own environment. Second, it aims to be in close proximity to policies associated to other environments. While the former is codified in the objective function of the CAL optimization problem, the latter appears as a linear constraint. The degree of proximity between policies is tuned by a parameter termed as the centrality measure. Our *second contribution* is to present properties of the optimizers of the CAL problem as the centrality measure varies from low to high values. We show that when this parameter is low, all policies are close to each other and so the obtained optimizers have good *generic performance*. That is, policies perform well across all environments. On the other hand, when the parameter value is high, each agent's policy maximizes performance in its own environment, that is, it displays good *specific performance*. Our *third contribution* is to identify conditions under which it is preferable for an agent to choose a policy obtained from the CAL problem over the individually optimal one. In particular, when an agent has inaccurate information about expert behavior and all environments and expert behaviors are sufficiently close, then the policy obtained from the CAL problem has a provably better performance as compared to the individual one. We formalize this statement by deriving a sufficient condition on the various error bounds. Since the CAL problem is nonconvex, our *fourth contribution* is an outer convex approximation of the problem using McCormick envelopes. We then discuss how this approximation can be solved in a distributed

manner. Our *last contribution* demonstrates the properties of the CAL framework in a numerical example where agents learn to navigate to a goal position in a windy gridworld.

We organize the rest of our article as follows. Section II provides preliminaries. The CAL framework is presented in Section III. In Section IV, we provide properties of the optimizers of the CAL optimization problem. Section V outlines a convex outer approximation of the CAL problem. Section VI demonstrates the use of the presented framework for a navigation task in a windy gridworld environment.

II. PRELIMINARIES

Here we collect notations and background on perturbation analysis of optimization problems.

A. NOTATIONS

We use $\mathbb{R}$ and $\mathbb{R}\geqslant 0$ to denote real and nonnegative real numbers, respectively. Unless otherwise specified, $\|\cdot\|$ is $\|\cdot\|_2$. By e_k we represent a vector of dimension N with all entries being 0 expect for the kth entry which is 1. A vector with all entries as unity is denoted by 1. A k-dimensional simplex is represented by $\Delta^k = \{x \in \mathbb{R}^k_{\geqslant 0} \mid 1^\top x = 1\}$. For any positive integer n, we use the notation $[n] = \{1, 2, \ldots, n\}$. The number of elements in a set $\mathcal{S}$ is denoted by $|\mathcal{S}|$. Given two sets X and Y, a set-valued map $f : X \rightrightarrows Y$ associates to each point in X a subset of Y. The set-valued map f is closed if its graph $\text{gph}(f) := \{(x, y) \in X \times Y \mid y \in f(x)\}$ is closed. Furthermore, the set-valued map f is upper semicontinuous at a point $x_0 \in X$ if for any neighborhood $\mathcal{N}_{f(x_0)}$ of the set $f(x_0)$ there exists a neighborhood $\mathcal{N}_{x_0}$ of x_0 such that for every $x \in \mathcal{N}_{x_0}$ the inclusion $f(x) \subset \mathcal{N}_{f(x_0)}$ holds. If this property holds for all $x_0 \in X$, then f is said to be upper semicontinuous.

B. PERTURBATION OF PARAMETERIZED OPTIMIZATION PROBLEMS

Consider the following problem:

$$\begin{aligned} \min_{x} \quad & f(x, u) \\ \text{subject to} \quad & x \in \mathcal{X}, \\ & G(x, u) \leqslant 0, \end{aligned} \tag{1}$$

where u is a parameter that belongs to a closed set $u \in \mathcal{U} \subset \mathbb{R}n_u$ and $\mathcal{X} \subset \mathbb{R}n_x$ is a closed set. The functions $f : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ and $G : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ are continuous. The feasibility set for the above optimization problem can be parameterized as the following set-valued map:

$$\mathcal{U} \ni u \mapsto \mathcal{H}(u) := \{x \in \mathcal{X} \mid G(x, u) \leqslant 0\}. \tag{2}$$

Similarly, the set of optimizers of (1) is written as the following set-valued map:

$$\mathcal{U} \ni u \mapsto \mathcal{S}(u) := \arg\min_{x \in \mathcal{H}(u)} f(x, u). \tag{3}$$

We are interested in the continuity of the map $\mathcal{S} : \mathcal{U} \rightrightarrows \mathcal{X}$ in the neighborhood of a point $u_0 \in \mathcal{U}$ (see Section II-A for relevant definitions).

Proposition 1: Upper semicontinuity of $\mathcal{S}$ [[25] Proposition 4.4] Given $u_0 \in \mathcal{U}$, suppose the following hold:

1) the map $\mathcal{H}(\cdot)$ is closed,
2) there exists $\alpha \in \mathbb{R}$ and a compact set $C \subset \mathcal{X}$ such that for every u in the neighborhood of u_0, the level set $\text{lev}_\alpha f(\cdot, u) := \{x \in \mathcal{H}(u) \mid f(x, u) \leqslant \alpha\}$ is nonempty and contained in C,
3) for any neighborhood $\mathcal{N}_{\mathcal{S}(u_0)} \subset \mathcal{X}$ of the set $\mathcal{S}(u_0)$, there exists a neighborhood $\mathcal{N}_{u_0} \subset \mathcal{U}$ of u_0 such that $\mathcal{N}_{\mathcal{S}(u_0)} \cap \mathcal{H}(u) \neq \emptyset$ for all $u \in \mathcal{N}_{u_0}$.

Then, the set-valued map $u \mapsto \mathcal{S}(u)$ is upper semicontinuous at u_0.

III. PROBLEM STATEMENT

We consider N learning agents and their corresponding environments. Each agent $i \in [N]$ is associated with a Markov decision process given by the tuple $\mathcal{M}_i = (\mathcal{S}, \mathcal{A}, P^i, \gamma, \nu_0)$. Here, the finite sets $\mathcal{S} := (s_1, \ldots, s_{|\mathcal{S}|})$ and $\mathcal{A} := (a_1, \ldots, a_{|\mathcal{A}|})$ represent the common state and action spaces, respectively. Agents evolve in different environments specified by their individual transition matrices. In particular, the transition matrix for agent i is $P^i \in [0, 1]^{|\mathcal{S}||\mathcal{A}| \times |\mathcal{S}|}$, where given a state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$, the row corresponding to it, $P^i_{(s,a),:}$, gives the distribution of the next state. For notational convenience, we also denote this distribution as $P^i(\cdot|s, a)$. Thus, given a state $\hat{s} \in \mathcal{S}$, the probability of reaching it from state s using action a is $P^i(\hat{s}|s, a)$. The discount factor and the distribution of the initial state of all agents are denoted by $\gamma \in (0, 1)$ and $\nu_0 \in \Delta^{|\mathcal{S}|}$, respectively.

An agent i has access to an expert's behaviour in its environment. Each expert i acts according to a policy given by the map $\pi_{E_i} : \mathcal{S} \to \Delta^{|\mathcal{A}|}$, that is, at state $s \in \mathcal{S}$, the distribution of the selected action by the expert is given by $\pi_{E_i}(s)$. We assume that each expert's policy is stationary and we denote the set of stationary policies by Π, that is, $\Pi := \{\pi : \mathcal{S} \to \Delta^{|\mathcal{A}|}\}$. The expert i's policy is aimed at minimizing a cost function $c_i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ associated with the task. This cost is unknown to us, however, the set where the cost belongs is known and is given as $c_i \in C_{\text{lin}} := \left\{\sum_{j=1}^{n_c} w_j \psi_j \mid \|w\|_\infty \leqslant 1\right\}$, where each w_j is the j-th component of the cost weight vector $w \in \mathbb{R}^{n_c}$ and it represents the weight associated to the j-th basis vector $\psi_j \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$. These n_c basis vectors are fixed and they satisfy $\|\psi_j\|_\infty \leqslant 1$ for all $j \in [n_c]$. Note that the cost basis vector are assumed to be common for all the environments. The behavior of the expert that is governed by its policy is known to us through the occupation measure that it generates. We elaborate on this next.

Given a policy π and the initial distribution $\nu_0 \in \Delta^{|\mathcal{S}|}$, the induced probability measure over the canonical sample space $\Omega := (\mathcal{S} \times \mathcal{A})^\infty$ for agent i is, $\mathbf{P}^{\pi,i}_{\nu_0}[\cdot]$. Here, $\mathbf{P}^{\pi,i}_{\nu_0}[s_t = s, a_t = a]$ denotes the probability that agent i is in state s and takes an

action a at time instant t starting from an initial state distribution ν_0 and following a policy π. For a given $\pi \in \Pi$, the discounted occupation measure for agent i, denoted $\mu_i^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, is defined as $\mu_i^\pi(s,a) := \sum_{t=0}^{\infty} \gamma^t \mathbf{P}_{\nu_0}^{\pi,i}[s_t = s, a_t = a]$. It is interpreted as the discounted expected number of times a state-action pair is visited by the agent i starting from an initial state distribution ν_0, and by following a policy π. We assume that for each environment i the occupation measure generated by the expert $\mu_i^{\pi_{E_i}}$ is known. This constitutes the behavior of the expert available to us. For environment i, consider the set

$$\mathcal{F}_i := \left\{ \mu \in \mathbb{R}_{\geqslant 0}^{|\mathcal{S}||\mathcal{A}|} \mid \left(B - \gamma P^i\right)^\top \mu = \nu_0 \right\}, \tag{4}$$

where $B \in \{0,1\}^{|\mathcal{S}||\mathcal{A}| \times |\mathcal{S}|}$ is a binary matrix where the element $B_{(s_j,a_k),s_l} = 1$ if $j = l$, and $B_{(s_j,a_k),s_l} = 0$ otherwise. From [3, Theorem 2] we know that, for every $\pi \in \Pi$, the corresponding occupation measure μ_i^π belongs to $\mathcal{F}_i$. Also, given any $\mu_i \in \mathcal{F}_i$, a stationary policy $\pi_{\mu_i} \in \Pi$ is obtained by setting $\pi_{\mu_i}(s,a) := \frac{\mu_i(s,a)}{\sum_{a' \in \mathcal{A}} \mu_i(s,a')}$. In addition, this correspondence is one-to-one, that is, the induced occupation measure for the policy π_{μ_i} is $\mu_i^{\pi_{\mu_i}} = \mu_i$. Given any cost function $c_i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, the expected discounted cost incurred by the agent i is $\eta_{c_i}(\pi) = \mathbf{E}_{\nu_0}^{\pi,i}\left[\sum_{t=0}^{\infty} \gamma^t c_i(s_t, a_t)\right]$, here the expectation is with respect to the distribution $\mathbf{P}_{\nu_0}^{\pi,i}$. This can also be represented as the inner product of discounted occupation measure and the cost vector, that is, $\eta_{c_i}(\pi) = \langle \mu_i^\pi, c_i \rangle$.

For agent i, the goal of learning (when decoupled from the other agents and environments) is to find a policy $\pi_i \in \Pi$ such that $\left\langle \mu_i^{\pi_i}, c_i \right\rangle \leqslant \left\langle \mu_i^{\pi_{E_i}}, c_i \right\rangle$, where $\mu_i^{\pi_{E_i}}$ is the occupation measure induced by the expert's policy π_{E_i}. However, note that the expert's cost c_i is usually unknown as only the behavior in terms of trajectories is available. Instead of knowing the exact cost, we assume that the agent knows the set C_{lin} where the true cost belongs. Consequently, the goal of learning then translates to finding a policy π_i such that $\langle \mu_i^{\pi_i}, c \rangle \leqslant \langle \mu_i^{\pi_{E_i}}, c \rangle$ for all $c \in C_{\text{lin}}$, i.e., the policy π must out-perform the expert policy for all $c \in C_{\text{lin}}$. Such a framework is well studied in the apprenticeship learning literature, see e.g., [4], [10], and [3]. Thus, the objective for agent i in apprenticeship learning, decoupled from all other agents and environments, is:

$$\min_{\pi_i \in \Pi} \sup_{c \in C_{\text{lin}}} \left(\left\langle \mu_i^{\pi_i}, c \right\rangle - \left\langle \mu_i^{\pi_{E_i}}, c \right\rangle \right). \tag{5}$$

One can simplify the objective function (5) by utilizing the structure of C_{lin}. Following the notation in [12], we define $\Psi := [\psi_1, \ldots, \psi_{n_c}] \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}| \times n_c}$ as the *cost basis matrix*. For every π_i, the following holds [12, Lemma 1],

$$\sup_{c \in C_{\text{lin}}} \left(\left\langle \mu_i^{\pi_i}, c \right\rangle - \left\langle \mu_i^{\pi_{E_i}}, c \right\rangle \right) = \left\| \Psi^\top \mu_i^{\pi_i} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1.$$

Thus, problem (5) can be equivalently written as

$$\min_{\pi_i \in \Pi} \left\| \Psi^\top \mu_i^{\pi_i} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1. \tag{6}$$

Note that the objective function is nonnegative and the optimal value is zero as $\pi_i^* = \pi_{E_i}$ is one of the optimizers. Additionally, for any optimizer π_i^* of (6), we have

$$\left\langle \mu^{\pi_i^*}, c \right\rangle = \left\langle \mu^{\pi_{E_i}}, c \right\rangle \quad \text{for all } c \in C_{\text{lin}}.$$

If C_{lin} contains all possible cost functions, then the expert policy is the only optimizer of (6). The lower the number of basis vectors in C_{lin} the more flexibility we have to find a policy that performs as well as the expert policy π_{E_i}.

Each agent i can solve problem (6) and obtain an optimal policy that performs well in its own environment. Such a policy might not perform well in other environments, while the learning task is same in all environments. To capture these commonalities between the environments, motivated by [17], we define the following *cross apprenticeship learning* (CAL) problem:

$$\min_{\{\pi_i\}_{i=1}^N, \pi_c} \quad \sum_{i=1}^{N} \left\| \Psi^\top \mu_i^{\pi_i} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1 \tag{7a}$$

$$\text{subject to} \quad \pi_c \in \Pi, \tag{7b}$$

$$\pi_i \in \Pi \text{ for all } i \in [N], \tag{7c}$$

$$\left\| \pi_i - \pi_c \right\|_\infty \leqslant \epsilon \text{ for all } i \in [N]. \tag{7d}$$

We denote the set of optimizers of the above problem by $\mathcal{S}^{\text{cal}} \subset \Pi^{N+1}$. In the above problem, through the decision variable π_i we seek a policy that performs well in environment i. The objective function is decoupled in this set of *individual policies*. On the other hand, these individual policies are required to be close to a *cross-learned* policy π_c. The variable ϵ defines this proximity and is termed as the *centrality measure*. The individual policies an agent learns via cross-learning sacrifices optimality in its environment for generalization across all other environments. Note that the objective function (7a) is linear in occupational measures and the constraints (7b)–(7d) are linear in the set of policies. Since the relationship between policies and occupational measures is nonlinear, the optimization problem is nonconvex. This is elaborated further in Section V. Regarding the objecitve function (7a), we can consider different cost basis vectors that are encoded in matrix Ψ for each environment without affecting the computational effort of solving the problem. Such a case can capture different form of tasks in each environment, for example.

Our aim in this paper is to analyze the properties of the CAL framework (7) and design methods to solve this optimization problem approximately.

IV. PROPERTIES OF CAL FRAMEWORK

The objective of this section is to analyze the performance of the individual and the cross-learned policies across different environments. We consider the following general *performance function* for a policy $\pi \in \Pi$:

$$V_\beta \pi := \sum_{i=1}^{N} \beta_i \left\| \Psi^\top \mu_i^{\pi} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1, \tag{8}$$

where $\beta \in \Delta^N$ represents the weight given to individual environments. In the above definition, the value $\left\| \Psi^\top \mu_i^\pi - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1$ determines how well the policy π performs in an environment i. A lower value indicates that the cost incurred by the policy is close to that by the expert. Therefore, a low value of performance function implies that the policy performs better across environments, where the importance attached to each environment is represented by the weighing β. Below we will analyze the properties of the above function.

A. CONTINUITY OF V_β

The right-hand side of (8) depends on the policy implicitly through the occupation measure generated. Therefore we will first examine the maps representing the correspondence between the policy and the occupation measure. To this end, we define the following two maps between the policy space Π and the set of feasible occupation measures $\mathcal{F}_i$ (see (4)) for some environment $i \in [N]$:

$$\mathrm{g}_i : \mathcal{F}_i \to \Pi, \text{ where } \mathrm{g}_i(\mu)(\mathrm{s}, \mathrm{a}) = \frac{\mu(\mathrm{s}, \mathrm{a})}{\sum_{\mathrm{a}' \in \mathcal{A}} \mu(\mathrm{s}, \mathrm{a}')} \tag{9a}$$

$$\mathrm{h}_i : \Pi \to \mathcal{F}_i, \text{ where } \mathrm{h}_i(\pi)(\mathrm{s}, \mathrm{a}) = \sum_{t=0}^{\infty} \gamma^t \mathbf{P}_{\nu_0}^{\pi,i}[s_t = s, a_t = a], \tag{9b}$$

for all $\mu \in \mathcal{F}_i$, $\pi \in \Pi$ and $(s, a) \in \mathcal{S} \times \mathcal{A}$. Note that g_i is same for all environments. We have used the subscript to denote that the domain is different for each of these functions. Recall that in the shorthand notation that we introduced earlier, we use $\mathrm{g}_i(\mu) = \pi_\mu$ and $\mathrm{h}_i(\pi) = \mu_i^\pi$. Before we delve into analyzing properties of the above defined maps, we derive the following bounds on the occupation measure that will be used later.

Lemma 2. (Bounds on the state occupation measure): Given a policy $\pi \in \Pi$ and any environment $i \in [N]$, for all $s \in \mathcal{S}$, it follows that $\nu_0(s) \leqslant \sum_{a \in \mathcal{A}} \mu_i^\pi(s, a) \leqslant \frac{|\mathcal{A}|}{1-\gamma}$.

Proof: Following the definition of the occupation measure, we have,

$$\begin{aligned}
\sum_{a \in \mathcal{A}} \mu_i^\pi(s, a) &= \sum_{a \in \mathcal{A}} \sum_{t=0}^{\infty} \gamma^t \mathbf{P}_{\nu_0}^{\pi,i}[s_t = s, a_t = a] \\
&= \sum_{t=0}^{\infty} \gamma^t \sum_{a \in \mathcal{A}} \mathbf{P}_{\nu_0}^{\pi,i}[s_t = s, a_t = a] \\
&= \sum_{t=0}^{\infty} \gamma^t \mathbf{P}_{\nu_0}^{\pi,i}[s_t = s] \\
&\overset{(a)}{=} \mathbf{P}_{\nu_0}^{\pi,i}[s_0 = s] + \sum_{t=1}^{\infty} \gamma^t \mathbf{P}_{\nu_0}^{\pi,i}[s_t = s] \overset{(b)}{\geqslant} \nu_0(s),
\end{aligned}$$

where $\mathbf{P}_{\nu_0}^{\pi,i}[s_t = s]$ is the probability that the agent i is in state s at time instant t starting with initial distribution ν_0 and following policy π. In the above relations, (a) is obtained by taking out the first term from the summation and (b) is due to that fact that the second term is nonnegative in the previous equality and $\nu_0(s) = \mathbf{P}_{\nu_0}^{\pi,i}[s_0 = s]$. For the upper bound we have,

$$\begin{aligned}
\sum_{a \in \mathcal{A}} \mu_i^\pi(s, a) &= \sum_{a \in \mathcal{A}} \sum_{t=0}^{\infty} \gamma^t \mathbf{P}_{\nu_0}^{\pi,i}[s_t = s, a_t = a] \\
&\leqslant \sum_{a \in \mathcal{A}} \sum_{t=0}^{\infty} \gamma^t = \frac{|\mathcal{A}|}{1-\gamma}.
\end{aligned}$$

This completes the proof. ■

Using the preceding results, in the following Lemma we present continuity properties of the maps g_i and h_i. In particular, both these functions are bijections, continuously differentiable, and Lipschitz.

Lemma 3. (Properties of the maps g_i and h_i): For some environment $i \in [N]$, consider the maps g_i and h_i as defined in (9). The following properties hold:

1) Maps g_i and h_i are continuously differentiable on $\mathcal{F}_i$ and Π, respectively.
2) For all $\mu_1, \mu_2 \in \mathcal{F}_i$ we have, $\|\mathrm{g}_i(\mu_1) - \mathrm{g}_i(\mu_2)\|_2 \leqslant \frac{2}{\min_{s \in \mathcal{S}} \nu_0(s)} \|\mu_1 - \mu_2\|_1$.
3) There exists a $L_i^{\mathrm{h}} > 0$ such that, for all $\pi_1, \pi_2 \in \Pi$ we have, $\|\mathrm{h}_i(\pi_1) - \mathrm{h}_i(\pi_2)\|_2 \leqslant L_i^{\mathrm{h}} \|\pi_1 - \pi_2\|_2$.

Proof: The map h_i has Lipschitz continuous gradient over the set Π, as shown in [26, Proposition 1]. Thus, h_i is continuously differentiable. Regarding g_i, denote the Jacobian as the map $D\mathrm{g}_i : \mathcal{F}_i \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}| \times |\mathcal{S}||\mathcal{A}|}$. For a given μ, the (i, j)-th element of the Jacobian $D\mathrm{g}_i(\mu)$, where index i and j correspond to state-action pairs (s_i, a_i) and (s_j, a_j), respectively, is

$$D\mathrm{g}_i(\mu)(i, j) = \begin{cases} \frac{\sum_{a' \in \mathcal{A}} \mu(s_i, a') - \mu(s_i, a_i)}{\left(\sum_{a' \in \mathcal{A}} \mu(s_i, a')\right)^2} & \text{if } (s_i, a_i) = (s_j, a_j), \\ \frac{-\mu(s_i, a_i)}{\left(\sum_{a' \in \mathcal{A}} \mu(s_i, a')\right)^2} & \text{if } s_i = s_j, \\ 0 & \text{otherwise.} \end{cases}$$

From Lemma 2 we know that for any state $s \in \mathcal{S}$, we have $\sum_{a' \in \mathcal{A}} \mu(s, a') \geqslant \nu_0(s) > 0$. Thus, $D\mathrm{g}_i$ given in the above expression is well-defined and continuous on $\mathcal{F}_i$. This proves the first claim. The second claim was established in [26, Proposition 1]. The last conclusion follows from the facts that h_i has a Lipschitz continuous gradient and it is continuously differentiable in Π. ■

The Lipschitz property of the map h_i established in the above result aids us in showing the same for the performance function V_β given in (8). The next result formalizes this statement. This property implies that if two policies are close to each other, as might be the case due to the centrality constraint (7d) in the CAL problem, then their performance across the environments will be similar.

Lemma 4. (Sensitivity of the performance function with respect to policies): Given two policies $\pi_1, \pi_2 \in \Pi$ and $\beta \in$

Δ^N, the following holds:

$$\left|V_\beta \pi_1 - V_\beta \pi_2\right| \leqslant n_c L^{\mathrm{h}} \sqrt{|\mathcal{S}|\,|\mathcal{A}|}\, \left\|\pi_1 - \pi_2\right\|_2 \tag{10}$$

where $L^{\mathrm{h}} = \max_{i\in[N]} L_i^{\mathrm{h}}$, with each L_i^{h} being the Lipschitz constant for the map h_i as stated in Lemma 3.

Proof: We compute

$$\begin{aligned}
\left|V_\beta \pi_1 - V_\beta \pi_2\right| &= \left| \sum_{k=1}^{N} \beta_k \Big(\left\| \Psi^\top \mu_k^{\pi_1} - \Psi^\top \mu_k^{\pi_{E_k}} \right\|_1 \right. \\
&\qquad \left. - \left\| \Psi^\top \mu_k^{\pi_2} - \Psi^\top \mu_k^{\pi_{E_k}} \right\|_1 \Big) \right| \\
&\overset{(a)}{\leqslant} \sum_{k=1}^{N} \beta_k \Big(\left\| \Psi^\top \mu_k^{\pi_1} - \Psi^\top \mu_k^{\pi_2} \right\|_1 \Big) \\
&\overset{(b)}{\leqslant} \left\|\Psi^\top\right\|_{1,1} \sum_{k=1}^{N} \beta_k \Big(\left\| \mu_k^{\pi_1} - \mu_k^{\pi_2} \right\|_1 \Big) \\
&\overset{(c)}{\leqslant} n_c L^{\mathrm{h}} \sqrt{|\mathcal{S}|\,|\mathcal{A}|}\, \left\|\pi_1 - \pi_2\right\|_2 \sum_{k=1}^{N} \beta_k \\
&\overset{(d)}{=} n_c L^{\mathrm{h}} \sqrt{|\mathcal{S}|\,|\mathcal{A}|}\, \left\|\pi_1 - \pi_2\right\|_2,
\end{aligned}$$

where inequality (a) is a consequence of the triangle inequality, (b) is due to the submultiplicity of induced matrix norm, (c) is due to Lemma 3 and the fact that the elements of the cost basis Ψ satisfy $\|\psi_j\|_\infty \leqslant 1$ for all $j = 1, \ldots, n_c$, and (d) is because $\beta \in \Delta^N$. ■

From the above result, by considering $\beta = e_i$ for some environment $i \in [N]$, we obtain the bound on the difference in the performance of two policies in that environment. This set of Lemmas will be useful in the subsequent section in analyzing the specific and generic performance of the policies obtained through the CAL problem.

B. SPECIFIC AND GENERIC PERFORMANCE OF CAL

As mentioned earlier, the solution of the CAL-framework results in N individual policies and a cross-learned policy. In this section, we investigate the performance of these policies in individual environments as well as across environments. We term these properties as *specific* and *generic* performance, respectively. We demonstrate how by tuning the centrality measure ϵ, one targets to maximize for one of these performances.

We first introduce relevant notation. Let the set of optimal policies for the decoupled learning problem of agent i given in (5) be denoted as $\mathcal{S}_i^{\mathrm{dec}} \subset \Pi$. That is,

$$\mathcal{S}_i^{\mathrm{dec}} := \arg\min_{\pi_i \in \Pi} \left\| \Psi^\top \mu_i^{\pi_i} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1.$$

We refer to $\pi_i^{\mathrm{dec},*} \in \mathcal{S}_i^{\mathrm{dec}}$ as the optimal *decoupled policy*. Note that if we choose ϵ to be large enough, then the CAL framework finds these decoupled optimal policies in the form of individual policies. Specifically, if

$$2\epsilon > \max_{i,j\in[N]} \left\{ \left\| \pi_i - \pi_j \right\|_\infty \,\middle|\, \pi_i \in \mathcal{S}_i^{\mathrm{dec}}, \pi_j \in \mathcal{S}_j^{\mathrm{dec}} \right\},$$

then any optimizer of CAL, denoted $\left(\{\pi_i^*\}_{i=1}^N, \pi_c^*\right) \in \mathcal{S}^{\mathrm{cal}}$, satisfies $\pi_i^* \in \mathcal{S}_i^{\mathrm{dec}}$ for all $i \in [N]$. That is, not all constraints of the CAL problem are binding. On the other hand, when $\epsilon = 0$, then $\pi_i^* \in \mathcal{S}^{\mathrm{cen}}$ for all $i \in [N]$ where

$$\mathcal{S}^{\mathrm{cen}} := \arg\min_{\pi \in \Pi} \sum_{i=1}^{N} \left\| \Psi^\top \mu_i^{\pi} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1. \tag{11}$$

We refer to any policy $\pi^{\mathrm{cen},*} \in \mathcal{S}^{\mathrm{cen}}$ as the optimal *centralized policy*. We have the following first result that provides the specific performance of the optimizers in $\mathcal{S}^{\mathrm{cal}}$.

Proposition 5. (Specific performance of $\mathcal{S}_i^{dec}$ and $\mathcal{S}^{cal}$): For any $\pi_i^{\mathrm{dec},*} \in \mathcal{S}_i^{\mathrm{dec}}$ and $\left(\{\pi_i^*\}_{i=1}^N, \pi_c^*\right) \in \mathcal{S}^{\mathrm{cal}}$, the following hold:

1) $V_{e_i}(\pi_i^{\mathrm{dec},*}) \leqslant V_{e_i}(\pi_i^*) \leqslant V_{e_i}(\pi_c^*)$, where e_i is the unit vector with i-th component being unity,
2) $V_{e_i}(\pi_i^*) \leqslant V_{e_i}(\pi_j^*)$ for all i and $j \neq i$.

Proof: The first inequality follows from the fact $\pi_i^{\mathrm{dec},*}$ is an optimizer of $\pi \mapsto V_{e_i}(\pi)$ over the set Π and π_i^* belongs to the set Π. For the second inequality, note that

$$\pi_i^* \in \arg\min_{\pi\in\Pi} \left\{ V_{e_i}(\pi) \mid \left\|\pi - \pi_c^*\right\|_\infty \leqslant \epsilon \right\}. \tag{12}$$

This is true because otherwise we contradict the fact that $\left(\{\pi_i^*\}_{i=1}^N, \pi_c^*\right)$ is an optimizer of (7). From the above observation and the fact that π_c^* trivially belongs to the feasibility set of (12), we conclude the second inequality. The last inequality also follows from the fact that π_j^* belongs to the feasibility set of (12), therefore $V_{e_i}(\pi_i^*)$ is at most equal to $V_{e_i}(\pi_j^*)$. ■

The above result shows that, as expected, the decoupled policy of environment i outperforms the individual and the cross-learned policy obtained from the CAL problem in that environment. Moreover, in environment i, the individual optimal policy π_i^* obtained in CAL performs better than the cross-learned policy π_c^* and any other individual policy π_j^*. The above result is irrespective of the value of the centrality measure. Next, we analyze the performance of the policies obtained across environments, where the selection of centrality measure ϵ becomes key. We will use Lemma 4 and show that for small enough values of ϵ, the individual optimal policies of CAL outperform the decoupled optimal policies across environments. In order to obtain the formal result, the first step is to analyze the set-valued map that gives the set of optimizers of the CAL problem given the parameter ϵ.

Lemma 6. (Upper semicontinuity of set of optimizers of CAL with respect to ϵ): Define the map,

$$\begin{aligned}
\Phi(\epsilon) := \Big\{ \Big(\{\pi_i\}_{i=1}^N, \pi_c \Big) \in \Pi^{N+1} \;\Big|\; \\
\left\|\pi_i - \pi_c\right\|_\infty \leqslant \epsilon, \text{ for all } i \in [N] \Big\}
\end{aligned} \tag{13}$$

that gives the feasibility set of (7) for a given $\epsilon \geqslant 0$. Then, the set-valued map $\mathcal{S}^{\text{cal}} : [0, 1] \rightrightarrows \Pi^{N+1}$ defined as

$$\mathcal{S}^{\text{cal}}(\epsilon) := \underset{\left(\{\pi_i\}_{i=1}^N, \pi_c\right) \in \Phi(\epsilon)}{\arg\min} \sum_{i=1}^{N} \left\| \Psi^\top \mu_i^{\pi_i} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1 \quad (14)$$

is upper semicontinuous at $\epsilon = 0$.

Proof: Our proof is based on Proposition 1 that analyzes the continuity of optimizers of a parameterized optimization problem. Drawing the parallelism between (3) and (14), the decision variable x, the parameter u, the set $\mathcal{U}$, the objective function f, and the set-valued map $\mathcal{H}$ as given in (3) are to be considered analogously in (14) as the variable $(\{\pi_i\}_{i=1}^N, \pi_c)$, the parameter ϵ, the set [0,1], the objective function

$$\bar{f}\left(\{\pi_i\}_{i=1}^N, \pi_c\right) := \sum_{i=1}^{N} \left\| \Psi^\top \mu_i^{\pi_i} - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1,$$

and the map Φ defined in (13), respectively. Note that in (14), the objective function does not depend on the parameter. The proof now proceeds by checking the conditions of Proposition 1. Firstly, the objective function $\bar{f}$ and the constraint function in (13) are continuous. The set-valued map Φ is closed. The second condition in Proposition 1 holds as $\bar{f}$ is bounded on Π^{N+1} and the set $\Phi(\epsilon)$ is nonempty and contained in the compact set Π^{N+1} for any nonnegative ϵ. Lastly, for the third condition, note that for every neighborhood $\mathcal{N}_0 \subset [0, 1]$ of $\epsilon = 0$, we have $\Phi(0) \subset \Phi(\epsilon)$ for all $\epsilon \in \mathcal{N}_0$. Consequently, $\mathcal{S}^{\text{cal}}(0) \subset \Phi(\epsilon)$ for all $\epsilon \in \mathcal{N}_0$ and so for any neighborhood $\mathcal{N}_{\mathcal{S}^{\text{cal}}(0)}$ of $\mathcal{S}^{\text{cal}}(0)$ we have $\mathcal{N}_{\mathcal{S}^{\text{cal}}(0)} \cap \Phi(\epsilon)$ for all $\epsilon \in \mathcal{N}_0$. Thus, condition three in Proposition 1 holds and so, we conclude that $\mathcal{S}^{\text{cal}}$ is upper semicontinuous at the origin. ■

With the above continuity property in mind, we next show that if ϵ is small, then the individual optimal policy obtained in CAL has better performance across environments as compared to the decoupled optimal policy.

Proposition 7. (Generic performance of $\mathcal{S}_i^{dec}$ and $\mathcal{S}^{cal}$): For any $\left(\{\pi_i^*\}_{i=1}^N, \pi_c^*\right) \in \mathcal{S}^{\text{cal}}$ and $\pi^{\text{cen},*} \in \mathcal{S}^{\text{cen}}$, we have

$$V_{N^{-1}\mathbf{1}}\pi^{\text{cen},*} \leqslant V_{N^{-1}\mathbf{1}}(\pi_i^*)$$

where $N^{-1}\mathbf{1}$ denotes the vector with each entry as $\frac{1}{N}$. Further, for any $j \in [N]$, if $V_{N^{-1}\mathbf{1}}(\pi^{\text{cen},*}) < V_{N^{-1}\mathbf{1}}(\pi_j^{\text{dec},*})$, then there exists an $\epsilon > 0$ such that the following holds,

$$V_{N^{-1}\mathbf{1}}\pi_j^* \leqslant V_{N^{-1}\mathbf{1}}(\pi_j^{\text{dec},*}) \text{ for all } (\{\pi_j^*\}_{j=1}^N, \pi_c^*) \in \mathcal{S}^{\text{cal}}(\epsilon).$$

Proof: The first inequality trivially follows from the definition of $\pi^{\text{cen},*}$. For the second inequality, from Lemma 4, the function $V_{N^{-1}\mathbf{1}}$ is Lipschitz continuous everywhere on the compact set Π. Thus, for any $i \in [N]$ where $V_{N^{-1}\mathbf{1}}(\pi^{\text{cen},*}) < V_{N^{-1}\mathbf{1}}(\pi_i^{\text{dec},*})$ holds, there exists a neighborhood $\mathcal{N}_{\mathcal{S}^{\text{cen}}} \subset \Pi$ of the set $\mathcal{S}^{\text{cen}}$ such that

$$V_{N^{-1}\mathbf{1}}\pi \leqslant V_{N^{-1}\mathbf{1}}(\pi_i^{\text{dec},*}), \quad \text{for all } \pi \in \mathcal{N}_{\mathcal{S}^{\text{cen}}}. \quad (15)$$

Noting the fact that $\mathcal{S}^{\text{cal}}(0) = (\mathcal{S}^{\text{cen}})^{N+1}$ and using Lemma 6, we conclude that there exists $\bar{\epsilon} > 0$ such that for all $\epsilon \in [0, \bar{\epsilon})$ we have $\mathcal{S}^{\text{cal}}(\epsilon) \subset (\mathcal{N}_{\mathcal{S}^{\text{cen}}})^{N+1}$. This inclusion along with the inequality (15) yields the conclusion. ■

The two results presented in this section highlight the fact that the optimizers of the CAL framework balance the properties of the centralized and decoupled optimal policies. This balance is tunable using the centrality measure ϵ. As an additional advantage of our framework, our next section illustrates how agents can learn from each other's experts.

C. ROBUSTNESS AGAINST OUTLIERS

Here we present a scenario where an agent can benefit from the proximity of policies imposed by the CAL framework. To this end, we require the following assumption that the occupancy measure induced by a policy in two different environments is bounded by the difference in transition kernels of these environments.

Assumption 8. (Sensitivity of occupation measure to the changes in transition matrix): There exists $\tilde{C} > 0$ such that given a policy $\pi \in \Pi$ and any two environments $i, j \in [N]$, we have

$$\left\| \mu_i^{\pi} - \mu_j^{\pi} \right\|_1 \leqslant \tilde{C} \left\| P^i - P^j \right\|_1.$$

The following result gives a bound on the different between the occupancy measure induced by two feasible policies of the CAL problem.

Lemma 9. (Properties of feasible points of CAL): Let $(\{\pi_i\}_{i=1}^N, \pi_c)$ be a feasible point of the CAL problem. Suppose Assumption 8 holds. Assume that $\|P^i - P^j\|_1 \leqslant \delta_2$ for all $i, j \in [N]$. Then, we have

$$\left\| \mu_i^{\pi_i} - \mu_j^{\pi_j} \right\|_1 \leqslant 2\,|\mathcal{S}|\,|\mathcal{A}|\,L^{\text{h}}\epsilon + \tilde{C}\delta_2, \quad \text{for all } i, j \in [N].$$

Proof: Note that

$$\begin{aligned}
\left\| \mu_i^{\pi_i} - \mu_j^{\pi_j} \right\|_1 &\leqslant \left\| \mu_i^{\pi_i} - \mu_j^{\pi_i} \right\|_1 + \left\| \mu_j^{\pi_i} - \mu_j^{\pi_j} \right\|_1 \\
&\leqslant \tilde{C} \left\| P^i - P^j \right\|_1 + \left\| \mu_j^{\pi_i} - \mu_j^{\pi_j} \right\|_1 \\
&\leqslant \tilde{C} \left\| P^i - P^j \right\|_1 + \sqrt{|\mathcal{S}|\,|\mathcal{A}|} L^{\text{h}} \left\| \pi_i - \pi_j \right\|_2 \\
&\leqslant \tilde{C}\delta_2 + 2\,|\mathcal{S}|\,|\mathcal{A}|\,L^{\text{h}}\epsilon,
\end{aligned}$$

where the first is the triangle inequality, the second uses Assumption 8, the third uses the Lipschitz property given in Lemma 3, and the last inequality is due the fact that for feasible points of CAL problem, we have $\|\pi_i - \pi_j\|_\infty \leqslant 2\epsilon$. ■

Our main result is given next. It identifies conditions under which an agent would prefer using policy obtained from the CAL problem over the optimal decoupled policy. Particularly, this captures the scenario where environments and expert behaviors are similar while one environment has a poor estimate of the expert occupation measure.

Proposition 10. (Robustness against error-prone expert occupation measures): Assume that $\|\mu_i^{\pi_{E_i}} - \mu_j^{\pi_{E_j}}\|_1 \leqslant \delta_1$ and $\|P^i - P^j\|_1 \leqslant \delta_2$ for all $i, j \in [N]$. Suppose Assumption 8

holds. Let $k \in [N]$ be the agent with error-prone expert occupation measure, that is, $\mu_k^{\text{err}} \in \mathcal{F}_k$ be the estimate of the expert occupation measure available to agent k and

$$\left\| \Psi^\top \mu_k^{\text{err}} - \Psi^\top \mu_k^{\pi_{E_k}} \right\|_1 \geqslant M.$$

Let $(\{\pi_i^{\text{err},*}\}_{i=1}^N, \pi_c^{\text{err},*})$ be the optimizer of the CAL problem, where $\mu_k^{\pi_{E_k}}$ is replaced with μ_k^{err}. Let $\pi_{k,\text{err}}^{\text{dec},*}$ be the optimal decoupled policy for agent k using the available estimate μ_k^{err} of the expert occupation measure. That is,

$$\pi_{k,\text{err}}^{\text{dec},*} \in \arg\min_{\pi_k \in \Pi} \left\| \Psi^\top \mu_k^{\pi_k} - \Psi^\top \mu_k^{\text{err}} \right\|_1. \tag{16}$$

If the following condition is satisfied

$$M > 4\,|\mathcal{S}|\,|\mathcal{A}|\,L^{\text{h}}\epsilon + 2\tilde{C}\delta_2 + 2\delta_1, \tag{17}$$

then $V_{e_k}(\pi_k^{\text{err},*}) < V_{e_k}(\pi_{k,\text{err}}^{\text{dec},*})$.

Proof: Due to optimality in (16), we have $\Psi^\top \mu_k^{\pi_{k,\text{err}}^{\text{dec},*}} = \Psi^\top \mu_k^{\text{err}}$. As a consequence, $V_{e_k}(\pi_{k,\text{err}}^{\text{dec},*}) = \|\Psi^\top \mu_k^{\pi_{k,\text{err}}^{\text{dec},*}} - \Psi^\top \mu_k^{\pi_{E_k}}\|_1 = \|\Psi^\top \mu_k^{\text{err}} - \Psi^\top \mu_k^{\pi_{E_k}}\|_1 \geqslant M$. For the sake of contradiction, assume that $V_{e_k}(\pi_k^{\text{err},*}) \geqslant V_{e_k}(\pi_{k,\text{err}}^{\text{dec},*}) \geqslant M$. Expanding this expression gives

$$\left\| \Psi^\top \mu_k^{\pi_k^{\text{err},*}} - \Psi^\top \mu_k^{\pi_{E_k}} \right\|_1 \geqslant M. \tag{18}$$

Note that for any $j \in [N]$, $j \neq k$, we have

$$\begin{aligned}
&\left\| \Psi^\top \mu_j^{\pi_j^{\text{err},*}} - \Psi^\top \mu_j^{\pi_{E_j}} \right\|_1 \\
&= \Big\| \Psi^\top \mu_j^{\pi_j^{\text{err},*}} - \Psi^\top \mu_k^{\pi_k^{\text{err},*}} + \Psi^\top \mu_k^{\pi_k^{\text{err},*}} - \Psi^\top \mu_k^{\pi_{E_k}} \\
&\qquad + \Psi^\top \mu_k^{\pi_{E_k}} - \Psi^\top \mu_j^{\pi_{E_j}} \Big\|_1 \\
&\geqslant \left\| \Psi^\top \mu_k^{\pi_k^{\text{err},*}} - \Psi^\top \mu_k^{\pi_{E_k}} \right\|_1 - \left\| \Psi^\top \mu_j^{\pi_j^{\text{err},*}} - \Psi^\top \mu_k^{\pi_k^{\text{err},*}} \right\|_1 \\
&\qquad - \left\| \Psi^\top \mu_k^{\pi_{E_k}} - \Psi^\top \mu_j^{\pi_{E_j}} \right\|_1 \\
&\geqslant M - 2\,|\mathcal{S}|\,|\mathcal{A}|\,L^{\text{h}} n_c \epsilon - \tilde{C} n_c \delta_2 - n_c \delta_1,
\end{aligned} \tag{19}$$

where in the above inequality, the first term is lower bounded due to (18), the second term is lower bounded due to Lemma 9 as both $\pi_k^{\text{err},*}$ and $\pi_j^{\text{err},*}$ are feasible points of the CAL problem, and the last term is lower bounded due to the hypothesis. Using the above inequality, we lower bound the value that the objective function of the CAL problem takes at the point $(\{\pi_i^{\text{err},*}\}_{i=1}^N, \pi_c^{\text{err},*})$ as

$$\begin{aligned}
&\left\| \Psi^\top \mu_k^{\pi_k^{\text{err},*}} - \Psi^\top \mu_k^{\text{err}} \right\|_1 + \sum_{j \neq k} \left\| \Psi^\top \mu_j^{\pi_j^{\text{err},*}} - \Psi^\top \mu_j^{\pi_{E_j}} \right\|_1 \\
&\geqslant (N-1)(M - 2\,|\mathcal{S}|\,|\mathcal{A}|\,L^{\text{h}} n_c \epsilon - \tilde{C} n_c \delta_2 - n_c \delta_1),
\end{aligned} \tag{20}$$

where the above uses (19). We next show that there exists a feasible point of the CAL problem that incurs less value than the right-hand side of the above inequality. This leads to a contradiction.

Consider a feasible point $(\{\pi_i\}_{i=1}^N, \pi_c)$ such that $\pi_k = \pi_c = \pi^{\mu_k^{\text{err}}}$. Then, the value the objective function of the CAL problem takes at this feasible point is

$$\begin{aligned}
&\sum_{j \neq k} \left\| \Psi^\top \mu_j^{\pi_j} - \Psi^\top \mu_j^{\pi_{E_j}} \right\|_1 \\
&\leqslant (N-1) \max_j \left\| \Psi^\top \mu_j^{\pi_j} - \Psi^\top \mu_j^{\pi_{E_j}} \right\|_1 \\
&= (N-1) \max_j \Big\| \Psi^\top \mu_j^{\pi_j} - \Psi^\top \mu_k^{\pi_k} + \Psi^\top \mu_k^{\pi_k} \\
&\qquad - \Psi^\top \mu_k^{\pi_{E_k}} + \Psi^\top \mu_k^{\pi_{E_k}} - \Psi^\top \mu_j^{\pi_{E_j}} \Big\|_1 \\
&\leqslant (N-1) \max_j \Big(\left\| \Psi^\top \mu_j^{\pi_j} - \Psi^\top \mu_k^{\pi_k} \right\|_1 \\
&\qquad + \left\| \Psi^\top \mu_k^{\pi_{E_k}} - \Psi^\top \mu_j^{\pi_{E_j}} \right\|_1 \Big) \\
&\leqslant (N-1) \max_j \Big(\left\| \mu_j^{\pi_j} - \mu_k^{\pi_k} \right\|_1 + \left\| \mu_k^{\pi_{E_k}} - \mu_j^{\pi_{E_j}} \right\|_1 \Big) \\
&\overset{(a)}{\leqslant} (N-1) \left(2\,|\mathcal{S}|\,|\mathcal{A}|\,L^{\text{h}} n_c \epsilon + \tilde{C} n_c \delta_2 + n_c \delta_1 \right) \\
&\overset{(b)}{<} (N-1)(M - 2\,|\mathcal{S}|\,|\mathcal{A}|\,L^{\text{h}} n_c \epsilon - \tilde{C} n_c \delta_2 - n_c \delta_1) \\
&\overset{(c)}{\leqslant} \left\| \Psi^\top \mu_k^{\pi_k^{\text{err},*}} - \Psi^\top \mu_k^{\text{err}} \right\|_1 + \sum_{j \neq k} \left\| \Psi^\top \mu_j^{\pi_j^{\text{err},*}} - \Psi^\top \mu_j^{\pi_{E_j}} \right\|_1,
\end{aligned}$$

where (a) is due to Lemma 9, (b) follows from the condition (17), and the last inequality is due to (20). The above reasoning implies that there exists a feasible point that takes value strictly less than the optimal value which is a contradiction. This completes our proof. ■

V. ALGORITHMS FOR SOLVING CAL PROBLEM

In this section, we investigate both centralized and distributed approaches to approximate the solution of the CAL problem (7). To this end, observe that the objective function in (7) is non-convex with respect to the policies but is convex with respect to the corresponding induced discounted occupation measures. Therefore, in line with the approach used in [12], we proceed to rewrite (7) in terms of the discounted occupation measure. This process results into a convex objective, but renders the constraints bilinear, as explained below. We handle the nonconvexity caused by such constraints by forming convex outer approximation of the feasibility set.

Recalling the set of feasible occupation measures given in (4) and the bijection between policies and occupation measures, we rewrite (7) equivalently as

$$\min_{\{\mu_i\}_{i=1}^N, \pi_c} \quad \sum_{i=1}^N \left\| \Psi^\top \mu_i - \Psi^\top \mu_i^{\pi_{E_i}} \right\|_1 \tag{21a}$$

$$\text{subject to} \quad \mu_i \in \mathcal{F}_i \text{ for all } i \in [N], \tag{21b}$$

$$\pi_c \in \Pi, \tag{21c}$$

$$\left|\frac{\mu_i(s,a)}{\sum_{a'\in\mathcal{A}}\mu_i(s,a')}-\pi_c(s,a)\right|\leqslant\epsilon \quad \text{for all } i\in[N],\, s\in\mathcal{S},\; a\in\mathcal{A}. \tag{21d}$$

The equivalence here refers to the fact that policies obtained from the optimal occupation measures of the above problem along with the cross-learned policy will be an optimizer of (7). The constraint (21d) can be written as $|\mu_i(s,a)-\pi_c(s,a)\sum_{a'\in\mathcal{A}}\mu_i(s,a')|\leqslant\epsilon\sum_{a'\in\mathcal{A}}\mu_i(s,a')$ and so it is bilinear in variables π_c and μ_i. Thus, the feasibility set of the above problem is nonconvex, in general. For computational ease, we use a set of linear inequality constraints to bound the nonconvex feasibility set that is formed by the bilinear constraint (21d). To this end, we make use of McCormick envelopes [27], [28]. Specifically, consider a bilinear constraint $z=xy$ for decision variables x, y, and z where the former two are further constrained as $x_l\leqslant x\leqslant x_u$ and $y_l\leqslant y\leqslant y_u$. Then, the McCormick envelope for the set

$$\{(x,y,z)\in\mathbb{R}^3 \mid z=xy,\, x_l\leqslant x\leqslant x_u,\, y_l\leqslant y\leqslant y_u\} \tag{22}$$

is the set consisting of four linear inequalities in place of the bilinear equality:

$$\left\{(x,y,z)\in\mathbb{R}^3 \;\middle|\; \begin{array}{l} z\geqslant x_l y+xy_l-x_l y_l,\\ z\geqslant x_u y+xy_u-x_u y_u,\\ z\leqslant x_u y+xy_l-x_u y_l,\\ z\leqslant x_l y+xy_u-x_l y_u,\\ x_l\leqslant x\leqslant x_u,\, y_l\leqslant y\leqslant y_u. \end{array}\right\} \tag{23}$$

In the definition of the above set, the first two inequalities are the so called underestimating convex functions, and the next two are overestimating concave functions. The set defined in (22) is a subset of that in (23). In the following, we make use of this procedure to form an outer approximation of (21d).

Let $\{y_i, w_i\}_{i\in[N]}$ be the new set of decision variables where $y_i\in\mathbb{R}^{|\mathcal{S}|}$ and $w_i\in\mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$. The variable $y_i(s)$ will take the value of $\sum_{a\in\mathcal{A}}\mu_i(s,a)$ and the variable $w_i(s,a)$ will play the role of $\pi_c(s,a)\sum_{a\in\mathcal{A}}\mu_i(s,a)$. Then, the bilinear constraint $w_i(s,a)=\pi_c(s,a)\sum_{a\in\mathcal{A}}\mu_i(s,a)$ will be replaced with four linear inequalities, similar to the way explained above. With these additional decision variables, we define the following McCormick relaxation of (21) as

$$\min_{\substack{\{\mu_i\}_{i=1}^N,\pi_c,\\ \{w_i\}_{i=1}^N,y_i}} \quad \sum_{i=1}^N\left\|\Psi^\top\mu_i-\Psi^\top\mu_i^{\pi_{E_i}}\right\|_1 \tag{24a}$$

$$\text{subject to} \quad \mu_i\in\mathcal{F}_i,\;\forall\, i\in[N], \tag{24b}$$

$$\pi_c\in\Pi,\,\{w_i\}_{i=1}^N\in\mathbb{R}_{\geqslant 0}^{|\mathcal{S}||\mathcal{A}|},\;\forall\, i\in[N], \tag{24c}$$

$$\{w_i\}_{i=1}^N\in\mathbb{R}_{\geqslant 0}^{|\mathcal{S}||\mathcal{A}|},\;\forall\, i\in[N], \tag{24d}$$

$$y_i(s)=\sum_{a\in\mathcal{A}}\mu_i(s,a),\;\forall\, i\in[N], \tag{24e}$$

$$\text{for all } i\in[N],\; s\in\mathcal{S},\; a\in\mathcal{A}:$$

$$\left|\mu_i(s,a)-w_i(s,a)\right|\leqslant\epsilon y_i(s), \tag{24f}$$

$$w_i(s,a)\geqslant\nu_0(s)\pi_c(s,a), \tag{24g}$$

$$w_i(s,a)\geqslant y_i(s)+\frac{|\mathcal{A}|}{1-\gamma}(\pi_c(s,a)-1), \tag{24h}$$

$$w_i(s,a)\leqslant y_i(s)+\nu_0(s)(\pi_c(s,a)-1), \tag{24i}$$

$$w_i(s,a)\leqslant\frac{|\mathcal{A}|}{1-\gamma}\pi_c(s,a). \tag{24j}$$

Here constraints (24g)–(24j) are obtained using the underestimators and over-estimators for $w_i(s,a)=\pi_c(s,a)y_i(s)$ along with the bounds $\nu_0(s)\leqslant y_i(s)\leqslant\frac{|\mathcal{A}|}{1-\gamma}$ and $0\leqslant\pi_c(s,a)\leqslant 1$ for the occupation measure and policy, respectively. The next result summarizes the guarantee of the above approximation.

Proposition 11. (Solutions of (24) as approximation of those of (7)): If $(\{\pi_i\}_{i=1}^N,\pi_c)$ is a feasible point of the CAL problem (7), then there exist $\{y_i, w_i\}_{i\in[N]}$ such that these variables along with $(\{\mu_i^{\pi_i}\}_{i=1}^N,\pi_c)$ together are feasible for (24). Consequently, the optimal value of (24) is a lower bound for the optimal value of (7).

Note that if $(\{\mu_i^*\}_{i=1}^N,\pi_c^*)$ is part of the optimizers of (24), then the obtained policies from these measures might not be feasible for the CAL problem (7). To obtain feasible policies, one can resort to one of the following two strategies:

1) Project all the policies $\{\pi_{\mu_i^*}\}_{i=1}^N$ obtained from (24) onto an $\epsilon-$ball (under the inf-norm) with its centre as the cross-learned policy π_c^*.
2) Project all the individual policies $\{\pi_{\mu_i^*}\}_{i=1}^N$ onto an ϵ-ball centred at the average policy $\frac{1}{N}\sum_{i\in[N]}\pi_{\mu_i^*}$.

The former gives more importance to π_c^*, while the later perceives that the obtained individual policies $\pi_{\mu_i^*}$ perform well and so their average is selected as an estimate of the cross-learned policy.

Remark 12. (An inner approximation approach): The McCormick relaxation described above forms an outer convex approximation of the feasibility set. One can also form an inner convex approximation by using the bound given in Lemma 2. Specifically, given any environment $i\in[N]$ and two occupation measures $\mu_1,\mu_2\in\mathcal{F}_i$, we have

$$\left\|\pi_{\mu_1}-\pi_{\mu_2}\right\|_2\leqslant\frac{2}{\nu_0^{\min}}\left\|\mu_1-\mu_2\right\|_1, \tag{25}$$

where $\nu_0^{\min}:=\min_{s\in\mathcal{S}}\nu_0(s)$ and π_{μ_1} and π_{μ_2} are policies corresponding to measures μ_1 and μ_2, respectively. This bound was obtained in [26, Proposition 1] and a closer look at the proof in there reveals that occupation measures need not be restricted to $\mathcal{F}_i$ for the bound to hold. In fact, if two measures μ_1,μ_2 belong to the set $\mathcal{J}$, where

$$\mathcal{J}:=\left\{\mu\in\mathbb{R}_{\geqslant 0}^{|\mathcal{S}||\mathcal{A}|}\mid \nu_0^{\min}\leqslant\sum_{a'\in\mathcal{A}}\mu(s,a')\leqslant\frac{|\mathcal{A}|}{1-\gamma}\text{ for all } s\in\mathcal{S}\right\},$$

then the bound (25) is satisfied. Further, for any vector $z \in \mathbb{R}^n$, we have $\|z\|_\infty \leqslant \|z\|_2$ and $\|z\|_1 \leqslant n\|z\|_\infty$. Using these bounds in (25), we obtain

$$\left\|\pi_{\mu_1} - \pi_{\mu_2}\right\|_\infty \leqslant \frac{2\,|\mathcal{S}|\,|\mathcal{A}|}{\nu_0^{\min}} \left\|\mu_1 - \mu_2\right\|_\infty,$$

for all $\mu_1, \mu_2 \in \mathcal{J}$. Note that $\mathcal{F}_i \subset \mathcal{J}$ for all $i \in [N]$. Using these facts, the convex inner approximation of (21) is

$$\min_{\{\mu_i\}_{i=1}^N, \mu_c} \quad \sum_{i=1}^N \left\|\Psi^\top \mu_i - \Psi^\top \mu^{\pi_{E_i}}\right\|_1 \tag{26a}$$

$$\text{subject to} \quad \mu_i \in \mathcal{F}_i,\ \forall i \in [N], \tag{26b}$$

$$\mu_c \in \mathcal{J}, \tag{26c}$$

$$\left\|\mu_i - \mu_c\right\|_\infty \leqslant \frac{\nu_0^{\min}}{2\,|\mathcal{S}|\,|\mathcal{A}|}\epsilon,\ \forall i \in [N]. \tag{26d}$$

Once an optimizer $(\{\mu_i^*\}_{i=1}^N, \mu_c^*)$ of the above problem is obtained, then the individual policies are $\{\pi_{\mu_i^*}\}_{i=1}^N$ and the cross-learned policy is $\pi_{\mu_c^*}$. As the size of state and action spaces appear in the denominator of the constraint (26d), this approximation is very conservative and often leads to infeasibility for large state and action spaces. •

Remark 13. (Distributed computation): For applications in the real world, we can envision the scenario where information or behavior of the expert is not available at one particular geographical location. For example, two individuals can be driving two different vehicles in two different geographical locations. In such a case, it is desirable to solve the CAL problem or its convex approximations in a distributed manner. By this we mean that the data about the expert behavior and the model of the environment remains as local information for an agent and is not shared with other agents. Under this information constraint, the convex approximations (24) and (26) both have structures that allow easy implementation of distributed algorithm. They both have objective functions as the summation of local functions and constraints that are local once a consensus constraint is added. For this case, either one can opt for primal-dual distributed algorithms or distributed alternating direction method of multiplier, see [29] for complete details. However, solving the bilinear problem (21) in a distributed manner is unexplored in the literature and we plan to pursue it in future. •

VI. SIMULATIONS

Here we illustrate the properties of the proposed CAL framework using a navigation task in a windy gridworld. Such an environment is often used to demonstrate the efficacy of reinforcement learning algorithms [1]. We consider four gridworlds, each of which consists of 7×10 cells (similar to [1, Example 6.5]), as depicted in Fig. 1. These four instances differ in the magnitude of the crosswind that is flowing from bottom to top. Each cell in the gridworld is a state of the environment. An agent in the gridworld aims to reach the target cell by taking at each time instance one of the four

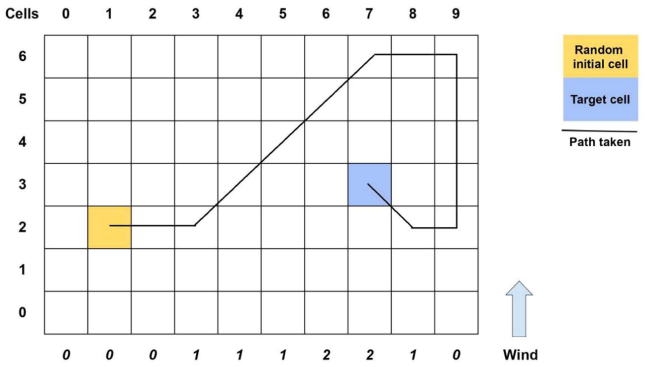

FIGURE 1. An instance of the windy gridworld and a sample trajectory of an agent in it. The yellow and the blue cells are the initial and target cells. The numbers at the bottom of each column stand for the magnitude of wind flowing in the upward direction in all cells belonging to that column.

available actions, i.e., move left, right, up, or down. When the magnitude of the wind at a particular cell is zero, then the action causes intended movement by one unit as long as it respects the boundary. For instance, action up results in moving of the agent by one unit in the upward direction. In case the wind has non-zero magnitude, then the displacement equivalent to the magnitude and along the direction of the wind is added to the displacement caused due to the action of the agent. For example, if the agent opts for moving right and the wind has unit magnitude, then the agent move to the top-right adjacent cell. This specifies completely the transition probability attached to an environment given the wind direction and magnitude at each cell. Roughly speaking, the aim for the agent is to reach the target cell (3,7), see Fig. 1, from any cell in the gridworld using minimum number of steps.

The direction of the wind for all environments and all cells is down to up. For each environment, the magnitude of the wind is the same for all cells in one column, refer to Fig. 1, and so the magnitude for the whole environment is specified by a vector. The wind vectors for four environments are:

Gridworld 1 : [0 0 0 1 1 1 2 2 1 0],

Gridworld 2 : [1 1 0 0 0 2 0 0 1 0],

Gridworld 3 : [0 1 0 1 2 0 1 1 1 0],

Gridworld 4 : [0 0 1 1 2 2 0 0 1 0].

One can observe a commonality to the task specified for each environment, while the transition probabilities differ. To obtain the behaviour of the expert specified by the occupation measure generated by the expert, we first obtain expert policies in each environment using ϵ-greedy SARSA algorithm, see ([1, Example 6.5]) for further details. Given the expert policies, we compute the discounted occupation measure generated by them using 10000 sample trajectories, each starting randomly at a location in the gridworld and consisting of 5000 time steps. For cost basis, we assume the simple case of $|\mathcal{S}||\mathcal{A}|$ number of vectors given by $\psi_i = e_i$ for all $i \in [|\mathcal{S}||\mathcal{A}|]$, where $e_i \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ has 1 at the ith position and all other entries are

TABLE 1 Average length of 1000 trajectories for which the policies obtained using $\epsilon = 1$, $\epsilon = 0.8$, $\epsilon = 0.6$, $\epsilon = 0.4$, $\epsilon = 0.2$ and $\epsilon = 0$ reached the target cell within a maximum of 2000 time steps.

Policy	World 1	World 2	World 3	World 4	Policy	World 1	World 2	World 3	World 4
		$\epsilon = 1$					$\epsilon = 0.8$		
Individual policy 1	12	1932	1818	1896	Individual policy 1	12	1930	1836	1892
Individual policy 2	1820	6	1842	691	Individual policy 2	1792	6	1830	659
Individual policy 3	1345	1483	8	1424	Individual policy 3	1350	1481	8	1399
Individual policy 4	1816	322	1870	6	Individual policy 4	1808	380	1872	6
Cross-learned policy	60	16	74	17	Cross-learned policy	65	15	68	16
		$\epsilon = 0.6$					$\epsilon = 0.4$		
Individual policy 1	15	193	92	205	Individual policy 1	19	23	29	27
Individual policy 2	1794	6	1828	10	Individual policy 2	1116	7	1061	8
Individual policy 3	42	41	10	47	Individual policy 3	1227	66	20	68
Individual policy 4	1790	8	1872	7	Individual policy 4	1742	8	1744	8
Cross-learned policy	87	18	92	19	Cross-learned policy	49	13	51	14
		$\epsilon = 0.2$					$\epsilon = 0$		
Individual policy 1	38	20	46	21	Individual policy 1	60	12	66	13
Individual policy 2	87	17	85	17	Individual policy 2	60	12	68	14
Individual policy 3	76	28	52	30	Individual policy 3	61	12	66	14
Individual policy 4	182	20	194	21	Individual policy 4	63	12	66	13
Cross-learned policy	69	19	76	21	Cross-learned policy	61	12	67	14

TABLE 2 Comparison between the cal optimization problem solutions using mccormick relaxation (24) solved using CVXPY with gurobi solver and directly solving (21) using an off-the-shelf non-convex solver gurobipy for the simulation example presented. Here, * represents the scenario where the solver fails to converge to the optimal value within 300 seconds, and only the lower and upper bound for the optimal value is returned.

	$\epsilon = 1.0$		$\epsilon = 0.8$		$\epsilon = 0.6$		$\epsilon = 0.4$		$\epsilon = 0.2$	
Solver	Value	Time (sec)	Value	Time (sec)	Value	Time (sec)	Value	Time (sec)	Value	Time (sec)
CVXPY	0.4318	0.1160	0.4318	0.1273	0.4330	0.1346	0.5126	0.1655	0.8846	0.2184
Gurobipy	0.4318	0.0751	0.4318	25.45	0.42-0.49*	300.21*	15.84-18.74*	302.91*	38.78-49.95*	301.99*

0. This completely specifies the CAL optimization problem that we aim to solve. We consider six values for the centrality measure, namely $\epsilon = 1$, $\epsilon = 0.8$, $\epsilon = 0.6$, $\epsilon = 0.4$, $\epsilon = 0.2$ and $\epsilon = 0$. We employ an outer approximation based on the McCormick envelope to find an approximate optimizer of the CAL problem. Since the obtained policies might not satisfy the closeness condition (7d), we use the second strategy explained in the discussion following Proposition 11 to obtain feasible cross-learned and individual policies. In Table 2 we present the difference between optimal values when solving the CAL problem without relaxation and with relaxation.

As stated earlier, the optimal value of the CAL objective increases as the coupling between agents' policies in different environments increases via a decrease in the value of ϵ, which can be observed in Fig. 2. The performance of the obtained policies is shown in Fig. 3 and Table 1. For each policy, we computed the occupation measure matching error between the expert occupation measure in an environment and the occupation measure induced by policies learned via the CAL framework. This error is indicated for each policy for each girdworld at different values of ϵ in Fig. 3. The closer the occupation measure of a policy in an environment is to the occupation measure of the expert in that environment, the better its performance will be. One can note that when ϵ is big, the individual policies are close to optimum in their respective environments (for $\epsilon = 1$ the agent learns its optimal decoupled policy) and their performance in other gridworlds is not necessarily good, e.g., Individual policy 3 for $\epsilon = 1$ in

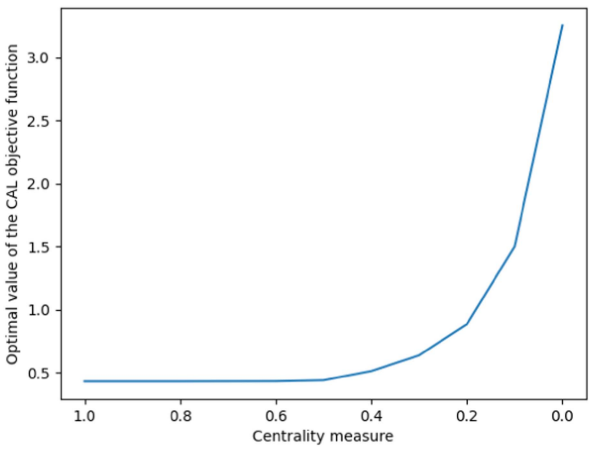

FIGURE 2. Optimal value of the CAL objective function solved via McCormick relaxation approach for different values of ϵ.

the Fig. 3. On the other extreme is the case of $\epsilon = 0$, and as observed from Fig. 3, all policies induce almost the same occupation measure matching error across environments for $\epsilon = 0$. Note that their performance may not exactly be the same as the obtained policies are stochastic, and we provide the results based on a finite number of trajectories. Our presented CAL framework balances these extreme cases when ϵ is chosen to be between 0 and 1. To present another approach for evaluating the performance of the policies learned

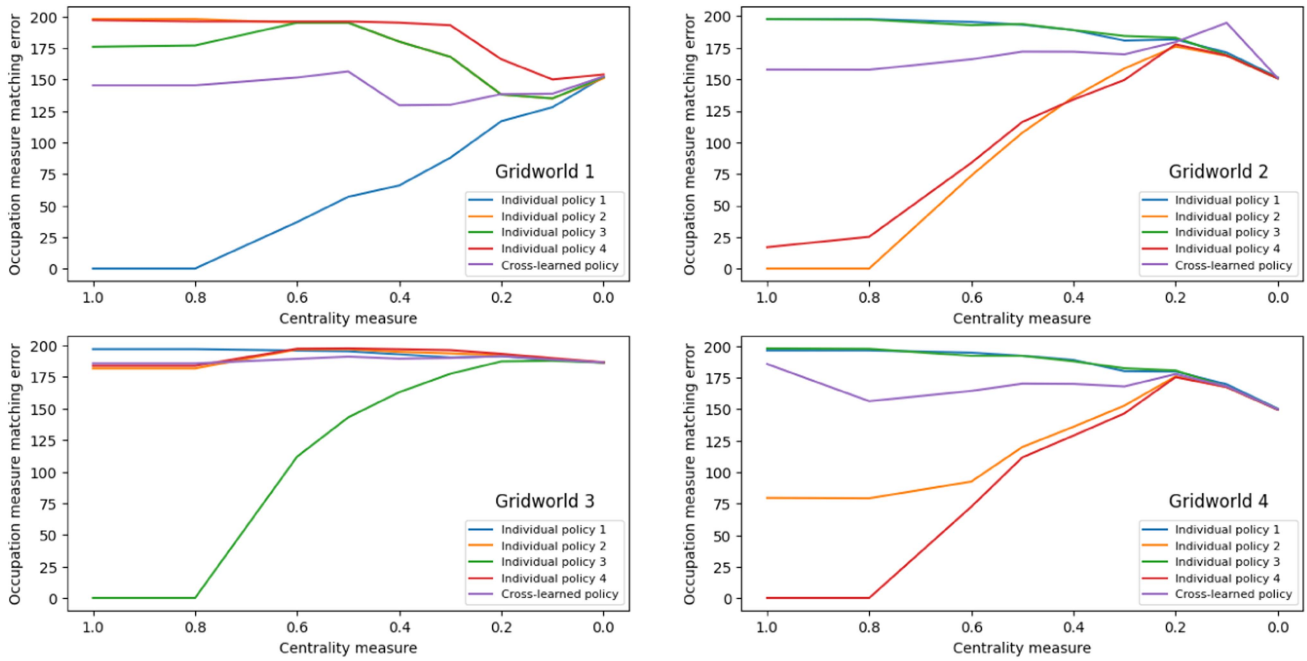

FIGURE 3. Variation in occupation measure matching error (calculated as 1-norm of the difference) between the expert occupation measure and occupation measures induced by policies learned via CAL framework in gridworlds 1, 2, 3 and 4 for different values of ϵ.

using the CAL framework, we provide Table 1 to indicate the average number of time steps taken by a policy to reach the goal state. Here the time steps are averaged over 1000 randomly initialized trajectories, and if the goal state is not reached within 2000 time steps for any trajectory, the time required to reach the target cell is taken as 2000. Observe that the average time steps required by various policies in an environment other than theirs decrease as we move from $\epsilon = 1$ to $\epsilon = 0$, indicating that the cross-learning has added to the performance of policies. However, there are some outliers to this general trend, possibly due to the non-convexity of the framework and approximation involved in finding the solution.

We next demonstrate two properties of our framework along the line of generalization and robustness. First, we investigate the performance of policies obtained using our above simulation setup on unseen environments to demonstrate how the CAL framework aids the generalization of learned policies experimentally. We denote them as Gridworld 5 and 6, with the following wind vectors:

$$\text{Gridworld 5}: \ [0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0]$$

$$\text{Gridworld 6}: \ [1\ 1\ 2\ 2\ 1\ 1\ 0\ 0\ 2\ 0].$$

Table 3 reports the performance in the sense of average time steps required to reach the target. It is observed from both tables that the cross-learned policy performs well in the unseen environments for all values of ϵ. On the other hand, in most cases, some individual policy performs poorly in unseen environments, but with the decrease in ϵ, the performance of the individual policies also improve. This indicates

TABLE 3 Average time steps required by the policies obtained using $\epsilon = 1$, $\epsilon = 0.8$, $\epsilon = 0.6$, $\epsilon = 0.4$, $\epsilon = 0.2$ and $\epsilon = 0$ to reach the target cell in two unseen environments within a maximum of 2000 time steps.

Policy	World 5	World 6	World 5	World 6
	$\epsilon = 1.0$		$\epsilon = 0.8$	
Individual policy 1	1880	459	1908	508
Individual policy 2	702	792	671	714
Individual policy 3	1420	1970	1411	1954
Individual policy 4	146	104	173	94
Cross learned policy	16	21	16	19
	$\epsilon = 0.6$		$\epsilon = 0.4$	
Individual policy 1	1503	277	26	25
Individual policy 2	9	9	8	8
Individual policy 3	46	1780	54	62
Individual policy 4	7	8	8	8
Cross learned policy	18	24	12	15
	$\epsilon = 0.2$		$\epsilon = 0.0$	
Individual policy 1	19	28	12	15
Individual policy 2	15	20	11	15
Individual policy 3	27	40	12	15
Individual policy 4	18	22	12	14
Cross learned policy	19	25	12	15

the ability of our framework to obtain policies that generalize well to unseen situations. Regarding the robustness of the CAL framework, we considered the scenario where the number of expert trajectories available in an environment is significantly lower than the number required to generate an accurate estimate of the expert occupation measure. In such a case, the individual and cross-learned policies may perform considerably better than the optimal decoupled policy in its environment. An example of such a scenario is presented in Table 4 where there is only a small number (10 and 15) of

TABLE 4 Average time steps required by the individual policy in environment 3 to reach the target cell when number of expert trajectories available were 10, 15 and 10000.

Number of expert Trajectories	10	15	10000
$\epsilon = 1.0$	186	42	8
$\epsilon = 0.9$	29	27	8
$\epsilon = 0.8$	8	27	8
$\epsilon = 0.7$	30	10	8
$\epsilon = 0.6$	52	34	10
$\epsilon = 0.5$	174	84	13

expert trajectories are available for environment 3. Observe from Table 4 that the performance of the individual policy learned by the agent in this scenario improves initially as the value of ϵ decreases. Moreover, for most of the considered ϵ values, the performance is better than that of the optimal decoupled policy, which corresponds to $\epsilon = 1$.

VII. CONCLUSION

We have introduced the cross apprenticeship learning (CAL) framework for apprenticeship learning when the expert trajectories of the task to be learned are available from multiple environments. We presented various properties of the optimizers of the problem that stands at the core of our framework. Further, since the problem is non-convex, we provided a convex approximation approach to solve it. Our findings were implemented in a numerical example related to navigation in a windy gridworld. Future work will explore distributed algorithms for bilinear optimization problems with tunable accuracy so as to solve the CAL problem for a large number of environments. We also wish to study agents' ability to learn from experts in other environments when the number of expert trajectories available is quite different in various environments. Lastly, we would like to explore the scalability of our approach to large-scale state-action spaces.

REFERENCES

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[2] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters, "An algorithmic perspective on imitation learning," *Found. Trends Robot.*, vol. 7, no. 1-2, pp. 1–179, 2018.

[3] U. Syed, M. Bowling, and R. Schapire, "Apprenticeship learning using linear programming," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 1032–1039.

[4] P. Abbeel and A. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2004, pp. 1–8.

[5] P. Abbeel, A. Coates, M. Quigley, and A. Ng, "An application of reinforcement learning to aerobatic helicopter flight," in *Proc. Adv. Neural Inf. Process. Syst.*, pp. 1–8.

[6] A. Coates, P. Abbeel, and A. Ng, "Learning for control from multiple demonstrations," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 144–151.

[7] J. Kolter, P. Abbeel, and A. Y. Ng, "Hierarchical apprenticeship learning with application to quadruped locomotion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 769–776.

[8] P. Abbeel, D. Dolgov, A. Y. Ng, and S. Thrun, "Apprenticeship learning for motion planning with application to parking lot navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 1083–1090.

[9] P. Fang, Z. Yu, L. Xiong, Z. Fu, Z. Li, and D. Zeng, "A maximum entropy inverse reinforcement learning algorithm for automatic parking," in *Proc. IEEE 5th CAA Int. Conf. Veh. Control Intell.*, 2021, pp. 1–6.

[10] U. Syed and R. E. Schapire, "A game-theoretic approach to apprenticeship learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1449–1456.

[11] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, NJ, USA: Wiley, 1994.

[12] A. Kamoutsi, G. Banjac, and J. Lygeros, "Stochastic convex optimization for provably efficient apprenticeship learning," in *Proc. Optim. Foundations Reinforcement Learn. Workshop, NeurIPS*, 2019.

[13] Y. A. Yadkori, P. L. Bartlett, X. Chen, and A. Malek, "Large-scale Markov decision problems via the linear programming dual," 2019, *arXiv:1901.01992*.

[14] P. Buchholz and D. Scheftelowitsch, "Computation of weighted sums of rewards for concurrent MDPs," *Math. Methods Operations Res.*, vol. 89, no. 1, pp. 1–42, 2019.

[15] S. Belogolovsky, P. Korsunsky, S. Mannor, C. Tessler, and T. Zahavy, "Inverse reinforcement learning in contextual MDPs," *Mach. Learn.*, vol. 110, no. 9, pp. 2295–2334, 2021.

[16] J. Chae, S. Han, W. Jung, M. Cho, S. Choi, and Y. Sung, "Robust Imitation Learning Against Variations in Environment Dynamics," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 2828–2852.

[17] J. Cervino, J. A. Bazerque, M. C. Fullana, and A. Ribeiro, "Multi-task reinforcement learning in reproducing Kernel hilbert spaces via cross-learning," *IEEE Trans. Signal Process.*, vol. 69, pp. 5947–5962, 2021.

[18] H. Yin, P. Seiler, M. Jin, and M. Arcak, "Imitation learning with stability and safety guarantees," *IEEE Control Syst. Lett.*, vol. 6, pp. 409–414, 2022.

[19] Z. Zhou, M. Bloem, and N. Bambos, "Infinite time horizon maximum causal entropy inverse reinforcement learning," *IEEE Trans. Autom. Control*, vol. 63, no. 9, pp. 2787–2802, Sep. 2018.

[20] S. Tesfazgi, A. Lederer, and S. Hirche, "Inverse reinforcement learning: A control Lyapunov approach," in *Proc. IEEE 60th Conf. Decis. Control*, 2021, pp. 3627–3632.

[21] S. Tu, A. Robey, T. Zhang, and N. Matni, "On the sample complexity of stability constrained imitation learning," in *Proc. Learn. Dyn. Control Conf.*, 2021, pp. 180–191.

[22] F. Memarian, A. Hashemi, S. Niekum, and U. Topcu, "Robust generative adversarial imitation learning via local lipschitzness," 2021, *arXiv:2107.00116*.

[23] L. J. Ratliff and E. Mazumdar, "Inverse risk-sensitive reinforcement learning," *IEEE Trans. Autom. Control*, vol. 65, no. 3, pp. 1256–1263, Mar. 2020.

[24] A. Kamoutsi, G. Banjac, and J. Lygeros, "Efficient performance bounds for primal-dual reinforcement learning from demonstrations," in *Proc. 38th Int. Conf. Mach. Learn.*, 2021, pp. 5257–5268.

[25] J. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*. Berlin, Germany: Springer, 2000.

[26] J. Zhang, A. Koppel, A. S. Bedi, C. Szepesvari, and M. Wang, "Variational policy gradient method for reinforcement learning with general utilities," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 4572–4583.

[27] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part I–convex underestimating problems," *Math. Program.*, vol. 10, pp. 147–175, 1976.

[28] F. A. Al-Khayyal and J. E. Falk, "Jointly constrained biconvex programming," *Math. Operations Res.*, vol. 8, pp. 273–286, 1983.

[29] G. Notarstefano, I. Notarnicola, and A. Camisa, "Distributed optimization for smart cyber-physical networks," *Found. Trends Syst. Control*, vol. 7, no. 3, pp. 253–383, 2019.